

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Aprendizado de redes neurais usando derivada fracionária de Caputo

Fernando Barbosa ¹João Batista Florindo ²

Departamento de Matemática Aplicada, Instituto de Matemática, Estatística e Computação Científica - Universidade de Campinas, Rua Sérgio Buarque de Holanda, 651, Cidade Universitária "Zeferino Vaz" - Distr. Barão Geraldo, CEP 13083-859, Campinas, SP, Brasil

1 Introdução

As redes neurais têm sido amplamente usadas tanto na indústria quanto na pesquisa acadêmica [1]. Em paralelo, em áreas da matemática e da física como cálculo tensorial, propagação de ondas e viscoelasticidade, o cálculo fracionário tem permitido a obtenção de resultados promissores, tendo sido usado para tratar de problemas envolvendo memória de um dado evento e algum nível de hereditariedade nos fenômenos.

Os resultados demonstrados por [2] justificam o uso do cálculo fracionário em redes neurais a partir de uma argumentação baseada na taxa de ativação dos neurônios e preservação de informações. Os autores sugerem, além de outras conclusões, que o parâmetro fracionário pode representar a capacidade do sistema neural humano de codificar frações da taxa de variação de informação realizada por um único neurônio.

2 Redes neurais de ordem α -fracionária

Entre os algoritmos de redes neurais fracionárias mais proeminentes, está o *backpropagation* usando a derivada de Caputo, com novas formulações baseadas em métodos de máxima descida. A convergência para a adaptação objeto deste estudo foi demonstrada conforme [2], assim como outros resultados relevantes para a rede neural, como o poder de generalização desta rede.

A derivada fracionária de Caputo de ordem $\alpha \in \mathbb{C}$ ($\text{Re}(\alpha) > 0$) utilizada nos experimentos é definida no intervalo real $[a, b]$. Quando $\alpha = n, n \in \mathbb{N}$, a n -ésima derivada de Caputo coincide com a n -ésima derivada usual para uma função real $y(x)$ definida em $[a, b]$, ou seja, $D_t^\alpha y(x) = y^{(n)}(x)$. A utilidade desta definição já é bem conhecida para a resolução de equações diferenciais com problema de valor inicial, mas também pode ser particularmente interessante em um algoritmo de aprendizado computacional.

¹fernando.liaison@gmail.com

²jbflorindo@ime.unicamp.br

3 Metodologia proposta

Na adaptação fracionária do *backpropagation* para redes neurais, a atualização dos pesos é feita por

$$w_{ij}^{k+1} = \frac{1}{(1-\alpha)\Gamma(1-\alpha)} w_{ij}^k (w_{ij}^k - c)^{1-\alpha}, \quad (1)$$

em que c é definido por

$$c = \min w_{i,j}^k. \quad (2)$$

Este processo de atualização é chamado de gradiente descendente [1]. Porém algumas melhorias sobre este algoritmo já são bem conhecidas, sendo uma das mais populares o gradiente descendente conjugado (GDC). Este traz algumas vantagens no que diz respeito à generalização do algoritmo e ao custo computacional em comparação com o método original.

O objetivo é adaptar o *backpropagation* de ordem fracionária para os otimizadores conjugados e avaliar o desempenho da generalização do algoritmo em problemas supervisionados de classificação. São estudados problemas em que dados de categorias distintas (imagens de objetos diferentes, por exemplo) são discriminados. Uma parte deste conjunto de dados tem sua categoria previamente conhecida (subconjunto de treinamento) e será usada para encontrar os parâmetros da rede, enquanto sobre a outra parte (teste) é que se realiza a classificação propriamente dita.

Vale destacar que o método do gradiente conjugado possui elementos que atuam como memória do processo de otimização. No cálculo fracionário, há também um reconhecido efeito de memória, cuja interação com as direções conjugadas do algoritmo GDC, o bi-conjugado BiGDC e outras variações, estão sendo estudadas neste trabalho. Os resultados obtidos até o momento apontam para uma taxa de acerto na classificação na base de dados *iris* da ordem de 96%, usando-se $\alpha = 7/9$, o que representa um avanço sobre o *backpropagation* clássico.

Agradecimentos

J.B.F. agradece o apoio financeiro da Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) (Processo nº 2016/16060-0) e do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) (Processo nº 301480/2016-8 e 423292/2018-8).

Referências

- [1] S. Haykin. Neural Networks. A comprehensive foundation. Pearson Prentice-Hall, 1999.
- [2] J. Wang, Y. Wen, Y. Gou et al. Fractional-order gradient descent learning of BP neural networks with Caputo derivative. *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 26, 2015.